[19] 中华人民共和国国家知识产权局

[51] Int. Cl⁷
G06F 9/46
H04L 29/06

[12] 发明专利申请公开说明书

[21] 申请号 00818837.8

[43] 公开日 2003年7月30日

[11] 公开号 CN 1433543A

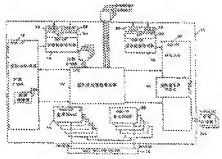
[22] 申请日 2000.12.21 [21] 申请号 00818837.8
[30] 优先权
[32] 2000.1.7 [33] US [31] 09/479,027
[32] 2000.1.7 [33] US [31] 09/479,028
[86] 国际申请 PCT/GB00/04950 2000.12.21
[87] 国际公布 WO01/50259 英 2001.7.12
[85] 进入国家阶段日期 2002.8.6
[71] 申请人 国际商业机器公司
地址 美国纽约
[72] 发明人 布莱恩·米切尔·巴斯
让·路易斯·卡尔威格纳
高登·泰勒·戴维斯
安托尼·马特奥·加罗
马克·海德斯
斯蒂芬·肯尼斯·詹金斯
罗斯·伯伊德·利文斯
迈克尔·斯蒂芬·西格尔
法布里斯·让·威尔布兰肯
[74] 专利代理机构 中国国际贸易促进委员会专利商标事务所
代理人 吴丽丽

权利要求书6页 说明书26页 附图25页

[54] 发明名称 帧和协议分类的方法和系统

[57] 摘要

数据处理(例如对数据分组或帧进行交换或路由选择)系统中的一种用于帧协议分类和处理的方法和系统。本发明包括根据预定的测试分析帧的一部分，然后存储分组的关键特征，用于随后的帧处理。帧(或输入信息单元)的关键特征包括：该帧中所用的第3层协议的类型，第2层压缩技术，起始指令地址，指示该帧是否使用虚拟局域网的标志，和该帧所属的数据流的身份。大多数分析最好利用硬件来实现，这样，可以快速地在统一的时间段内完成分析。然后，网络处理器合体在帧的处理过程中可以使用所存储的帧的特征。处理器用起始指令地址和第3层标题的起点位置以及帧类型的标志进行了预处理。也就是说，根据帧的类型，处理器利用该指令地址或代码入口点在合适的位置开始对帧进行处理。还可以依次堆积一些附加的指令地址并在分支处依次使用，以免进行附加测试和分支说明。此外，构成数据流的帧还可以按它们被接收时同样的次序被处理和转发。



ISBN 1008-4274

1. 一种设备，包括：

一个半导体衬底；

构造于该衬底上的 N 个处理单元，其中 N>1；

构造于所述衬底上的第一内部数据存储器，所述数据存储器用于存储可由所述 N 个处理单元访问的信息；

一个操作上与 N 个处理单元连接的分配器，用于接收输入信息单元和将输入信息单元发送到 N 个处理单元之一；

一个与分配器连接的分类器，所述分类器包括一个比较单元，用于确定输入信息单元的数据格式，和用于产生输入信息单元的输出指示符以及该输入信息单元的起始地址并将它们存储于内部数据存储器中，其中所述输出指示符指示该输入信息单元的数据格式，这些指示符和起始地址在 N 个信息处理单元之一处理输入信息期间对该 N 个处理单元之一可用，并用于输入信息单元的处理过程中；以及

位于所述半导体衬底上并且在操作上与 N 个处理单元相连的完成单元，用于接收由 N 个处理单元之一处理的信息单元。

2. 如权利要求 1 所述类型的设备，其特征在于：比较单元包括一种对输入信息数据中所含的虚拟局域网字段的测试，而所产生的输出指示符包括一种用于标识输入信息单元中存在虚拟局域网字段的标识符。

3. 如权利要求 1 或 2 的设备，其特征在于：分类器包括构造于衬底上的多个硬件装置。

4. 如权利要求 1、2 或 3 所述类型的设备，其特征在于：输出指示符包括一些用于标识输入信息单元的类型及其第 2 层的压缩技术的指示符。

5. 如权利要求 1、2、3 或 4 所述类型的设备，其特征在于：指示符包括缺省代码入口点。

6. 如权利要求 1 至 5 任一所述类型的设备，其特征在于：分类器包括一种用于确定基于由分类器所确定的输入信息单元的类型及其压缩技术的代码入口点的系统。

7. 如权利要求 1 至 6 任一所述类型的设备，其特征在于：分类器包括一种用于确定缺省代码入口点和基于输入信息单元的类型的代码入口点的系统。

8. 如权利要求 7 所述类型的设备，其特征在于：该设备还包括一个用于从缺省代码入口点和基于输入信息单元的类型的代码入口点中作出选择的选择器。

9. 如权利要求 8 所述类型的设备，还包括一种用于标识虚拟局域网信息是否包含在输入信息单元中的系统。

10. 如权利要求 1 至 9 任一所述类型的设备，其特征在于：指示符还包括被分配了输入信息单元的 N 个处理单元之一的身份。

11. 如权利要求 10 所述类型的设备，还包括一种标志，用于指示处理后的信息单元要按所述处理后的信息单元被接收时的次序从衬底发出，其中所述完成单元响应该标志以便当 N 个处理单元之一完成对信息单元的处理时分配所述处理后的信息单元。

12. 如权利要求 11 所述类型的设备，其特征在于：分类器包括一种系统，用于产生指示其数据流的各输入信息单元的标识符并将其存储在内部数据存储器中，和用于将来自同一数据流的后续信息单元与来自同一数据流的较早信息单元链接，其中处理器中较早的信息单元被标记为特定数据流中的第一信息单元，并且，来自处理单元的信息单元的传输局限于被标记为该特定数据流的第一信息单元的那些信息单元。

13. 如权利要求 1 至 12 任一所述类型的设备，其特征在于：分配单元还依次将各信息单元的标识符以及为进行处理而被分配了该信息单元的处理单元的身份存储于一个队列中；并且，其中，

完成单元还与该顺序队列连接，并利用分配器所分配的各信息单元的标识符以及被分配了该信息单元的处理单元的身份，以便按与信

息单元被接收时的次序相同的次序来组合处理后的信息单元。

14. 如权利要求 13 所述类型的设备，其特征在于：该设备还包括一种信号，用于覆盖该次序并按信息单元被完成的次序将处理后的信息单元发送到网络。

15. 一种处理输入信息单元的方法，包括以下步骤：

在分配器处接收输入信息单元；

将输入信息单元从分配器发送到多个处理器之一以进行处理；

当将信息单元从分配器发送到多个处理器之一时，从输入信息单元中读取选定的比特；

对照已知的标识输入信息单元的预定类型的指示符测试来自输入信息单元的读取比特，以标识输入信息单元的类型和协议，或者标识输入信息单元不符合输入信息单元的任何预定标识类型；和

根据来自输入信息单元的比特的测试的结果，存储输入信息单元的类型的指示符以及关于输入信息单元的其他信息；和

在多个处理单元之一中，在进行输入信息单元的处理时，将利用所存储的指示符和其他所存储的关于输入信息单元的信息。

16. 包括权利要求 15 的步骤的方法，其特征在于：产生指示符和存储指示符的步骤在输入信息单元正被发送到多个处理器之一时出现，这样，当多个处理器之一处理输入信息单元时，指示符和其他信息已被确定和存储，并且该多个处理器之一在处理该输入信息单元时可利用这些指示符和关于输入信息单元的其他信息。

17. 包括权利要求 15 或 16 的步骤的方法，还包括根据输入信息单元的内容产生用于进一步处理输入信息单元的起始地址的步骤，并且在多个处理单元之一中利用所存储的指示符的步骤包括利用起始地址。

18. 包括权利要求 15、16 或 17 的步骤的方法，其特征在于：读取和测试的步骤用硬件来完成，据此，这一过程可以以比通过执行一系列存储的指令来完成读取和测试更少的处理周期来完成。

19. 包括权利要求 18 的步骤的方法，其特征在于：用硬件来完成

输入信息单元的类型的标识和存储这些指示符的步骤在两个机器周期内完成，据此，在多个处理单元之一中利用指示符的步骤会比通过按次序执行一系列编程的指令来完成测试时更早出现。

20. 如权利要求 19 所述类型的处理信息单元的方法，其特征在于：测试和存储输入信息单元的指示符的步骤与在多个处理单元之一中接收来自分配器的输入信息单元的步骤相互重叠。

21. 包括权利要求 15 至 20 任一的步骤的方法，还包括如下步骤：

产生和存储输入信息单元的指示符；

与该标识符相关联地存储被分配了该信息单元的处理单元的身份；和

利用该标识符和被分配了信息单元的处理单元的身份，按信息单元被接收的次序来发送处理后的信息单元。

22. 包括权利要求 21 的步骤的方法，还包括通过将无标记标志包含于帧中来响应处理单元所产生的帧的步骤。

23. 如权利要求 22 所述类型的处理信息单元的方法，其特征在于：系统通过将帧传送到网络而不作进一步存储来响应无标记标志的帧。

24. 一种用于标识输入帧和为进一步处理帧而提供与该帧有关的指示符的方法，这种方法的步骤包括：

通过将输入帧的一部分与表示压缩类型和协议类型的预定内容进行比较，从输入帧确定压缩类型和协议类型；

与各输入帧相关地产生和存储该输入帧的压缩类型和协议类型的指示符；

确定和存储输入帧的第 3 层标题的位置；和

根据所确定的协议类型和压缩方法，确定和存储进一步处理该输入帧的起点，据此，在进一步处理该输入帧时可使用所述位置和起点。

25. 包括权利要求 24 的步骤的用于确定输入帧的特性的方法，其特征在于：确定用于进一步处理的起点的步骤包括如下步骤：从输入帧确定缺省代码入口点，然后，利用输入帧协议和压缩方法来确定针对压缩和协议的组合是否已存储了所存储的控制入口点，如果是，那

么用所存储的控制入口点作为进一步处理的起始点，否则，用缺省的代码入口点作为进一步处理的起始点。

26. 一种用于从网络中接收并处理不同格式的数据分组的设备，包括：

多个处理器，各处理器都相互独立地进行操作，用于处理数据分组和提供基于输入数据分组的输出数据分组；

一个与这些处理器连接的分配单元，用于接收来自网络的数据分组和将分组分配给多个独立处理器之一；

一个与分配单元连接的分类装置，用于接收分组和确定其协议和压缩技术以及处理单元进一步处理帧的起始地址，该分类装置包括：

根据帧的一部分来确定压缩技术的逻辑；

确定在帧中存在虚拟局域网信息的逻辑；和

包括压缩类型和用于进一步处理的起始地址的各帧的输出。

27. 如权利要求26所述类型的设备，其特征在于：分类器包含在硬件中，而没有被存储的程序。

28. 如权利要求26所述类型的设备，其特征在于：分类发生在两个周期内，这样，处理单元之一可以在帧被分配单元分配后的两个周期内开始对帧的进一步处理。

29. 如权利要求26所述类型的设备，其特征在于：起始地址这样来确定，即从帧中生成缺省起始地址，并且使用该缺省地址作为用于处理帧的起始地址，除非针对分类系统所确定的压缩方法和协议存储了不同的起始地址。

30. 如权利要求26所述类型的设备，其特征在于：该设备还包括一种处理系统，用于处理不是从网络接收到的新信息单元，这种处理系统给新信息单元添加一个记号，指示这种新信息单元不是从网络接收到的。

31. 一种设备，用于分析具有可变协议和压缩的信息帧以及用于提供用于处理该帧的起始位置和到处理该帧的初始指令的指针，该设备包括：

一个比较器，用于查看帧的预定字节并判断这些字节是表示长度还是协议；

用于确定该帧的协议和压缩系统的逻辑；

利用协议和压缩系统来确定用于处理该帧的起始位置和到用于处理该帧的初始指令的指针。

帧和协议分类的方法和系统

技术领域

本发明涉及用于将各种类型和性能的信息处理系统或计算机链接在一起的通信网络设备，并涉及这种设备中的用于数据处理的部件和方法。具体地说，本发明涉及一种用于管理与数据传输网连接的处理设备中的数据流的改进型系统和方法，其中包括了一种用于处理多个输入信息单元（也称为“分组”或“帧”）的方法和系统，这些输入信息单元可以同时被多个独立处理器所处理，并且这些输入信息单元可以具有多种不同协议之一。

背景技术

本实施方式与下列文献有关，所有这些文献都已转让给本发明的受让人。

专利申请系列号 09/384,691，它由 Brian Bass 等人于 1999 年 8 月 27 日申请，名称为“网络处理器处理集合体和方法”（Network Processor Processing Complex and Methods），本文中该专利有时称为“网络处理单元专利”或“NPU 专利”。

美国专利 5,724,348，它于 1998 年 3 月 3 日发布，名称为“数据交换机的有效硬件/软件接口”（Efficient Hardware/Software Interface for a Data Switch），该专利有时称为“接口专利”。

专利申请系列号 09/330,968，它于 1999 年 6 月 11 日申请，名称为“数据通信的高速并行/串行链路”（High Speed Parallel/Serial Link for Data Communications），该专利有时称为“链路专利”。

转让给 IBM 的用于其多协议交换业务的各种专利和申请，这些专利和申请有时称为“MSS”，其中某些专利和申请包括 Cedric Alexander 作为发明人，并且有时称为“MSS 专利”。

以下对本实施方式的描述基于这样一个假设：读者具有关于网络数据通信以及这种网络通信中所用的路由器和交换机的基本知识。具体地说，这一描述假定读者熟悉将网络运行划分为层的网络体系结构的国际标准化组织（ “ISO”）模型。基于ISO模型的典型体系结构从作为信号向上所通过的物理通路或媒体的第一层（有时称为“L1”）至第2层（或 “L2”）、第3层（或 “L3”），等等，一直扩展到作为驻留在与网络链接的计算机系统中的应用编程层的第7层（或 “L7”）。在整个本文中，引用诸如L1、L2、L3等层旨在表示网络体系结构的相应层。本文的描述还假定读者基本了解网络通信中所用的比特串（称为分组或帧）。

在如今的网络运行中，对带宽的考虑（即系统在单位时间内能处理的数据量）正变得越来越重要。近年来，主要是随着因特网（松散链接的计算机的公用网，有时称为 “万维网”）的飞速发展，在较小程度上也随着专用数据传输网或内部网的普遍增长，网络业务量急剧增加。因特网和内部网涉及了远端之间的大量信息的传输，以满足日益增长的对信息和新出现的应用的远程访问需求。因特网已对地理上分散的地区的大量用户开放了大量的远程信息，并使得可以进行各种新的应用（比如电子商务）。它导致网络负荷越来越大。诸如电子邮件、文件传送和数据库访问等其他应用进一步增加了网络的负荷。由于网络业务量居高不下，有些应用已处于过度使用状态。

网络上的业务还越来越多样化。某些网络原来主要用于某种通信业务，比如，电话网上的话音和数据传输网上的数字数据。当然，除了话音信号外，电话网还可以载送一定量的 “数据”（比如主叫号码和被叫号码，用于选择路由和记帐），但某些网络的基本用途有时实际上已属于同类分组。

然而，目前，话音和数据业务正不断地聚合到相同的网络中。随着因特网的继续扩展及在可靠性和安全性等方面的技术的不断提高，已有可能相对同时地发送许多不同类型的信息，包括不同类型信息的混合体，比如，话音和数据。

目前，数据可以不计费地通过因特网（通过因特网协议即IP）进行传输，而话音业务通常也走费用最低的通路。诸如基于IP的话音（VoIP）和基于异步传送方式即ATM的话音（VoATM）或基于帧中继的话音（VoFR）等一些技术是当今环境条件下传输话音业务的合算备选方法。随着这些业务的变迁，产业将应对诸如改变费用结构的问题，以及对处理器之间信息传输的服务费用与服务质量进行折衷选择的考虑。

服务质量方面包括容量或带宽、响应时间（处理一帧需要多长时间）以及处理的灵活性（是否响应不同的协议和帧结构，比如不同的压缩或帧标题方式）。使用资源的人将根据情况折衷考虑服务质量和服务费用。

发送数据分组的一些现有技术的系统要求分组具有单一的协议或格式，或者具有所允许的有限种协议或帧中的一种。由于这种系统可是为这些所允许的协议所定制的，因此，当在系统中发现只有一种协议（或者有限种协议）的分组时，由于设计相对简单，这种系统具有速度和响应快的优点。当整个数据传输系统在单个实体控制下时，该控制实体易于强制用户使用单一标准传输协议（用户要么遵循所允许的协议要么不使用该网络，因为，网络被编程为只提供指定的协议而并不能处理这些协议的变形，即使是看起来较小的变形）。

然而，即使是来自通信“标准”如以太网的帧也可以利用几种协议之一被格式化，并且可以利用不同的压缩技术被压缩成一个消息。这些不同的协议和压缩技术通常在帧的起点处和在其他关键信息（如L3消息的起点）之前提供了不定量的数据。因此，来自以太网的帧的关键信息（如果有的话）可以根据以太网L3协议或以太网的形式和压缩技术位于该帧中的不同位置。对L3消息进行处理的系统需要首先找到它，在多协议系统中这可能是个艰巨的任务。因此，例如，以太网DIX版本2不同于以太网802.3，基于以太网的IPX不同于本身有三种不同格式（Novell专用、LLC和SNAP）的基于以太网802.3的IPX。此外，IPX的每种版本利用所谓的IEEE 802.1q标准可能支持也可能

不支持虚拟 LAN（即 VLAN），这还会改变帧的格式，从而改变 L3 消息的位置。

在那些现有技术的支持多种协议的帧的系统中，有时需要提供大量的开销（比如计算机编程，有时含有一百多行带有比较和分支指令的代码），用来识别协议和用来将帧从一种协议转换为另一种协议，或用来从帧中去除不必要的信息（比如压缩信息）。这种多协议处理过程是很耗时的，而且识别协议的时间量通常还是不定的。当这种系统需要不定量的时间来识别协议和提供必要的处理时，该系统不得不被配置为允许最长必要时间（以便处理最坏的情况），这样，所有帧的处理速度减慢到最坏情况，或者出现在分类所允许的时间内某些帧未被处理的可能性。

大多数处理器都从指令集的公共起点（对所有数据都在同一位置）开始进行处理，并设置这样一些标志，当处理器需要判断转到哪里和执行哪些指令时，它将有选择地读取这些标志。因此，许多处理器的执行都将完成一些测试，以判断它具有哪种数据以及从哪里开始进行实质性处理，这些测试涉及多个周期且可能涉及多个处理。

在现有技术中，已知一些用于处理数据的多处理器系统，这些系统采用了严格的先进先出的数据处理方式。虽然处理以常规方式进行时这种系统工作得很好，但是，当某一输入的处理被延迟时，这种系统将受约束并停止工作。某一输入的处理的延迟还会停止其他输入的处理。

还已知其他一些现有技术的系统，这些系统在处理期间一直对输入消息单元进行跟踪。这些系统的局限性和缺点在于：必须用大量的处理能力来一直跟踪每个信息单元在系统中的何处，并且某些系统并不提供附加的输入信息单元，比如来自新数据流或来自内部产生的消息的信息单元。

因此，用于处理数据分组的现有技术的系统有一些不合乎需要的缺点和局限性，这些缺点和局限性不是影响了系统的多功能性就是影响了系统运行的速度，或者这两者都受影响。对熟练技术人员而言，

纵观本发明的下列描述，还将看到现有技术的系统的其他缺点和局限性。

发明内容

本发明的实施方式可以克服现有技术的系统的这些缺点和局限性，它提供一种简单而又有效的管理在网络上的帧或分组的数据流的方法，这些帧或分组是利用众多不同的所允许的消息协议之一所构成的，并且它们可能使用也可能不使用虚拟局域网（即 VLAN）系统。通过快速有效地分析每一分组或帧，可以确定和保存该帧的帧类型和关键特征，以便将来例如在以上所引用的 NPU 专利中所述类型的网络处理器中用作参考和对该帧进行处理。

本实施方式的优点在于，它可以快速有效地处理具有不同协议的分组，并提供了更快更容易的分组处理方法，从而使得整个系统可以以高速的帧处理速率运行。

本实施方式使得路由器或交换机可以处理不定格式的连续分组或帧，而无需事先知道具体的帧或分组是以哪种格式构成的。这一实施方式包括：识别该消息或分组的第 2 层（L2）压缩格式，然后运用所存储的规则来识别 L2 压缩、L3 协议和虚拟局域网（VLAN）的存在。作为这种判定的结果，处理器可以在起始指令地址开始运行；也就是说，处理器用基于该帧的标识的指令的起始地址进行了预处理。因此，处理器具有起始指令地址和该帧的数据部分中的 L3 标题的起点的指针以及一些指示协议、VLAN 存在和压缩格式的标志。

本实施方式的优点在于，它在分组的初始处理期间建立并存储了关于分组的关键信息，然后，所存储的关于分组或帧的信息以后可以有利于例如 NPU 专利中所述的网络处理单元集合体在处理过程中所使用，从而使得在其后续阶段可以更快更有效地处理分组。

本实施方式的优点在于，它使得可以将来自单一数据流的输入分组或帧分配给用于处理的多个独立处理器之一，然后，使得可以将输

出（处理后的）分组或帧重组成与输入分组或帧被接收到的次序相同的次序。

本实施方式的优点在于，多个数据流可以互不影响地被处理，并且一个数据流不会阻塞其他数据流。也就是说，当一个数据流的处理因等待完成其某一部分的处理而停止时，其他数据流的处理可以继续。

本实施方式还可以使系统的刷新或所完成的帧的即刻分配与次序无关（需要的话），从而不考虑按所接收到的次序处理每一数据流的正常操作。

本实施方式的优点还在于，它便于缓冲器和存储装置的有效使用，并且它还行快捷，这样，处理速度不会因为管理数据流的开销而下降。

本实施方式预期它可以在与一组网络处理器及其相应的存储器件相同的半导体衬底上实现，从而使得在这些器件之间可以实现快速数据传输。

本实施方式还可以用硬件而不用软件来实现，并且，所需的格式测试可以在统一的时间内完成，而与确定格式或压缩技术之前的格式和必须进行多少次比较无关。在所示的设计中，在两个时钟周期内，可以完成帧的分类，同时设置必要的指示符来指示存在哪种类型的帧（例如采用了哪种压缩技术和哪种第3层协议）和是否支持虚拟LAN（即VLAN）以及关于该帧的关键信息。在这两个周期内，分配器可以将帧发送到空闲的网络处理单元（如所引用的NPU专利中所述）。作为判断协议和压缩方法的帧处理结果，可以确定处理器所需用的起始地址并将其传送给处理器，这样，由于处理器预装了起始地址（到有关的指令存储器的指针）以及其处理所需的其他有关的信息，从而它可以开始对该帧的工作。处理器预装其处理所需的起始地址有时称为处理器的预处理（preconditioning），这可使处理器更有效——它不必处理完许多测试指令和根据测试结果跳过一些指令，而是在所出现的消息的具体格式的初始地址处开始起动。

本实施方式的系统的优点还在于，帧的分类和预处理可以与将该帧分配到网络处理器合体中并行地进行。这种并行处理使得帧的处理更有效并可以使系统运行得更快。

通过使用本实施方式，多处理单元既可以相互独立，又可以处理同一数据流，而不会使其一部分陷入不同的和所不希望的次序。给定数据流的处理后的分组或帧的输出次序将与系统从数据流接收输入分组或帧的次序相同，除非由于刷新命令而被覆盖。

最后，本实施方式还允许系统插入新数据流和创建分组或帧，而不会影响保持从网络接收到的数据流的次序的处理过程。

本实施方式的一种增强型方式使得不仅可以进行处理器的预处理（存储第一指令的地址），而且还可以存储以后执行的一些指令的附加地址。这样，处理器不仅有第一指令的地址，而且还有后续分支（即分叉）点处的指令的地址，从而在代码的执行中避免了不必要的测试（如满足某条件，则转向指令#1，否则转向指令#2）。这使得代码的执行更有效。

对熟悉相关技术的人员而言，纵观优选实施方式的下列描述，并参照附图和附属权利要求书，还将看到本实施方式的其他目的和优点。

附图说明

因此，在陈述了现有技术的一些局限性和缺点以及本发明的一些目的和优点后，对熟悉相关技术的人员而言，纵观一种改进型路由选择系统和方法的用于说明本发明的附图的下列描述，还将看到其他目的和优点，其中：

图 1 是一个含有 NPU 专利中所述的实施本发明所用的嵌入式处理器集合体的接口设备的框图；

图 2 是一个图 1 中所示类型的嵌入式处理器集合体的框图，它具有一个本发明中有用的分类器硬件辅助；

图 3A-3T 是一些说明在本发明的硬件分类器中所使用的各种以太网协议格式的图解；

图 4 是本发明的分类器硬件辅助的流程图，用于说明本发明中的分类器处理帧部分所用的逻辑电路；

图 5 是一个用于说明本发明的分类器的功能图解；

图 6 是本发明的具有所示可选增强型器件的硬件分类器的备选实施方式，它除了第一指令的地址外还可使一系列地址存储在堆栈中；

图 7 是与每一帧相应的队列的示意图；

图 8 是本发明的完成单元的详细示图，其中 N 个处理器中的每个处理器都有两个标记存储器；

图 9 是一个用于一直对 N 个处理器中的每个处理器正在处理的数据流进行跟踪的标记存储器的格式的示意图；

图 10 是说明完成单元在接收和处理一个已将新的帧分配给处理单元之一的指示时所执行的逻辑的流程图；

图 11 是说明完成单元在处理一个帧处理已完成的报告时所执行的逻辑的流程图；和

图 12 是图 8 的完成单元的另一示图，其中包括了用以说明其优选实施方式中完成单元的操作的数据。

具体实施方式

在优选实施方式的以下描述中，将描述发明人目前所知的实施本发明的最佳实现方式的某些特性。不过，这一描述只是以一种特定实施方式中讲述本发明的大致的一般性的思想，而并不是将本发明局限于本实施方式中所示的情况，尤其对于相关技术的熟练人员而言，他们将认识到针对这些图所说明和描述的具体结构和操作可以有许多变形和更改。

图 1 是一个适合于与数据传输网连接的处理系统的功能框图，该处理系统用于以分组或信息单元（有时也称为帧，本文中将交替使用这些术语）的形式接收、处理数据以及将其重发给网络。如图 1 中所示，这种用于数据处理的系统包括了多个子组件，如 NPU 专利中所述，这些子组件有利地集成在单一衬底上。整个组件在单一衬底上集成便

于紧密封装系统的各个部件，从而减少了部件之间通信所需的时间，并因此提高了系统运行的速度。多处理器以及支持逻辑和存储器使用单一衬底还可以减少因相互连接而导致的故障的发生率，并且还可以提高对可能干扰网络中的数据传输的噪声或其他杂散信号的抵抗力。

装在衬底 10 上的子组件分为上部配置和下部配置，其中“上部”配置（有时也称为“入口”）是指与从数据传输网进入芯片（到达或进入芯片）的数据有关的那些部件，而“下部”（有时也称为“出口”）是指以离开方式（离开芯片或向下进入网络）将来自芯片的数据发向数据传输网的那些部件。数据流分别流向上部和下部配置；因此，在图 1 的系统中，存在着上部数据流和下部数据流。上部或入口配置单元包括排队-出队-调度 UP（EDS-UP）逻辑电路 16、多个复用的 MAC's-UP（PMM-UP）14、交换数据移动器-UP（SDM-UP）18、系统接口（SIF）20、数据调整串行链路 A（DASL-A）22 和数据调整串行链路 B（DASL-B）24。数据链路在以上所引用的链路专利中作了详述，因此应当参照这一文献，以便更好地理解系统的这一部分。应当理解，本发明的优选实施方式使用了如该专利中所详述的数据链路，但因为本发明并不局限于那些特定的辅助设备（比如本优选实施方式中所用的数据链路），其他系统也可以很好地应用本发明，尤其是那些支持相对高速的数据流和系统要求的系统。

本系统的下部（或出口）中所述的部分包括数据链路 DASL-A 26 和 DASL-B 28、系统接口 SIF 30、交换数据移动器 SDM-DN 32、排队-出队－调度器 EDS-DN 34 和出口的多个复用的 MAC PMM-DN 36。衬底 10 还包括多个内部静态随机存取存储器件（S-RAM）、一个业务量管理调度器（TRAFFIC MGT SCHEDULER）40 和一个如所引用的 NPU 专利中所详述的嵌入式处理器集合体 12。接口设备 38 通过各自的 DMU 总线与 PMM 14、36 连接。接口设备 38 可以是用于与 L1 电路系统连接的任意合适的设备，比如，以太网物理（ENET PHY）设备或异步传送方式的成帧设备（ATM FRAMER），这两种设备均是商业上众所周知的很适用于这一目的的设备的例子。接口设备的类

型和大小至少部分地由所用芯片及其系统所连接的网络媒体来确定。多个外部动态随机存取存储器件（D-RAM）和一个 S-RAM 可由该芯片所使用。

尽管这里特别针对这样的网络进行了讨论，即：其中，在相关交换和路由选择设备外部的一般数据流是通过导体（比如安装在建筑物内部的电线或电缆）传送的，然而，本发明预期网络交换机及其部件也可以应用于无线环境中。例如，这里所讨论的媒体访问控制（MAC）单元可以代之以合适的射频器件，比如，用硅锗技术制成的器件，这样可以使所讨论的器件直接与无线网连接。当适当地采用了这种技术时，对所属领域技术人员而言，可以将射频单元集成到这里所讨论的 VLSI 结构中。或者，射频或其他无线响应器件（比如红外（IR）响应器件）可以安装在带有这里讨论的其他单元的叶片（blade）上，以实现适用于无线网络设备的交换设备。

箭头表示图 1 中所示接口系统中的一般的数据流向。离开 ENET PHY 块 38 经过 DMU 总线从以太网 MAC 14 接收到的数据的帧或消息被 EDS-UP 器件 16 置于内部数据存储缓冲器 16a 中。这些帧被识别为正常帧或引导帧，这会涉及多个处理器中的后续处理的方法和位置。

图 2 是一个可以很好地应用本发明的处理系统 100 的框图。在图 2 中，多个处理单元 110 位于分配器单元 112 与完成单元 114 之间。每一到来帧 F（来自未示出的与本数据处理系统连接的网络）被接收后存储到通过接口 UP DS i/f 117 与处理单元 110 连接的 UP 数据存储器 116 中，所述接口 UP DS i/f 117 可以读取数据并将数据写到数据存储器中。这些帧依次被分配器 112 所移动并被分配给多个处理单元 110 之一，分配时根据分配器 112 判断该处理单元是否可以处理帧的情况来确定。这一指示可以是被分配了帧 F 的一个处理单元已将信号发送给分配器 112，表明该特定的处理单元有空执行任务，不过，在本系统中也可以很好地采用其他分配任务的方法（比如，循环分配法或最近所用算法）。关于具体的处理单元 110 的结构和功能和通常的处理系统的详尽描述，可以参见以上 NPU 专利参考文献。介于分配器 112

与多个处理单元 110 之间的是一个硬件分类器辅助 118，它将在本文中稍后具体结合图 4 和 5 作详细描述。与多个处理单元 110 相关的还有指令存储器 122（如图 4 中所示），在该存储器中，存储有众多不同的指令集，以供各处理单元 110 所检索和执行。如后面所述，根据基于硬件分类器辅助 118 所确定的消息类型（其协议和压缩方法）的地址，可以对指令存储器 122 中的起始指令寻址。

完成单元 114 操作上连接在多个处理器 110 与向下排队系统（标记为 "DN 排队"，即图 1 中的单元 34）以及 UP 排队系统（图 1 中的单元 16）之间。DN 排队系统 34 用来将来自处理集合体的处理后的帧向下发送到与该集合体连接的网络或其他系统中，而 UP 排队系统 16 用来将处理后的帧发送到交换结构中。分配器 112 可以用来分配和存储与各帧有关的和与被分配来处理这样的帧的处理单元有关的标识信息。于是，完成单元 114 可以用这种标识信息来确保处理后的构成单一数据流的帧按它们被接收到时的次序被转发。本发明的这一方面稍后将在本说明书中作更详细的讨论。

图 3（包括其各种分图，图 3A-3T）示出了本处理系统被编程来接受和处理的多种消息格式（以太网消息格式的组成部分和变形），不过，为了适合所考虑的系统的环境，消息或帧的格式的表是熟练技术人员可以改变的表。本系统还可以被重新设计成可以接受其他消息格式，包括将来可能指定的消息格式和变形。这样，为了描述帧的不同格式，图 3 的消息格式具有不同的协议和压缩类型，并且本发明是一个灵活系统，它被设计成可以接受各种不同的协议和压缩格式，并可以通过提供指向压缩和协议的类型的指针为这些帧的处理提供帮助，还可以为处理所给帧的处理器提供指令存储器中的起始地址。

图 3A 示出了一般的或基本的以太网消息格式。这种格式有时称为以太网版本 2.0/DIX。这种消息格式的消息包括：目标地址 DA，源地址 SA，指示消息类型的块（类型），消息正文即数据，和用于消息完整性检验的循环冗余校验即 CRC 的尾部。目标地址 DA 和源地址 SA 都被规定为 6 个字节（48 比特），指示 "类型"的块被规定为 2 个

图3R

802.3 MAC标题
DA
SA
长度
6字节
6字节
2
字节
LLC LPDU
SNAP
DSAP=
AA
SSAP=
AA
控制=
03
oui=
000000
EType=
0800
1字节
1字节
1字节
3字节
2字节
IPv4标题
如果是UDP
或TCP
vers
len
业务
类型
总
长度
标识
标志
标志
偏移
TTL
协议
校验和
源IP地址
目标IP
地址
可选项/
填充
源端口
目标
端口
4
比特
4
比特
1
字节
2
字节
2
字节
3
比特
13
比特
1
字节
1
字节
2
字节
4
字节
4
字节
0-n
字节
2
字节
2
字节

图30

802.3 MAC帧头
DA
SA
长度
6字节
6字节
2
字节
LLC LPDU
DSAP=
06
SSAP=
06
控制=
03
1字节
1字节
1字节
IPv4标头
如果是ICMP
就是ICMP
version
业务
类型
总
长度
标识
标志
片偏移
TTL
协议
校验和
源IP地址
目标IP
地址
可选项/
填充
源端口
目标
端口
4
比特
4
比特
1
字节
2
字节
3
比特
13
比特
1
字节
1
字节
2
字节
4
字节
4
字节
0~n
字节
2
字节
2
字节

图3P

以太网MAC标题
DA
SA
0800
6字节
6字节
2
字节
IPv4标题
vers
hlen
服务
类型
总
长度
标识
标志
碎片
偏移
TTL
协议
校验和
源IP
地址
目标IP
地址
可选项/
填充
如果是UDP
或TCP
源端口
目标
端口
4
比特
4
比特
1
字节
2
字节
2
字节
3
比特
13
比特
1
字节
1
字节
2
字节
4
字节
4
字节
0-n
字节
2
字节
2
字节

图30

802.3 MAC标题
DA
SA
8100
6字节
6字节
2
字节
VLAN分组
TCI
2字节
LEN/
ETYPE
2字节
CFI≠1
CFI=1
E-R1F
控制
2字节
LTH<=2
LTH>2
E-R1F
控制
2字节
(LTH-2)/2次
LLC LPDU
SNAP
DSAP=
AA
SSAP=
AA
控制=
03
oui=
000000
Etype=
8137
1字节
1字节
1字节
3字节
2字节
IPX标题
校验和
分组
长度
TC
PT
目标网络
目标节点
目标
套接字
源网络
源节点
源套
接字
2
字节
2
字节
1
字节
1
字节
4
字节
6
字节
2
字节
4
字节
6
字节
2
字节

图3N

802.3 MAC标题
DA
SA
8100
6字节
6字节
2
字节
VLAN分组
TCI
LEN/
ETYPE
GFI=1
E-RIF
控制
2字节
2字节
2字节
GFI=1
LTH<=2
LTH>2
E-RIF
描述符
2字节
(LTH-2)/2次
LLC LPDU
DSAP=
E0
SSAP=
E0
控制=
03
1字节
1字节
1字节
IPX标题
校验和
分组
长度
TC
PT
目标网络
目标节点
目标
插口
源网络
源节点
源插口
2
字节
2
字节
1
字节
1
字节
4
字节
6
字节
2
字节
4
字节
6
字节
2
字节

图3M

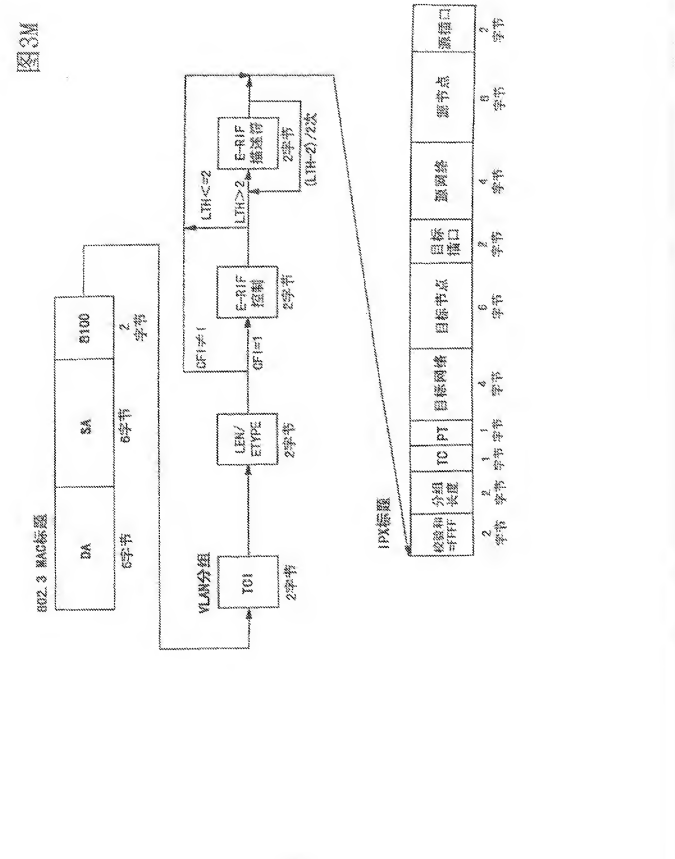

802.3 MAC标题
DA
SA
8100
6字节
6字节
2
字节
VLAN分组
TCI
2字节
LEN/
ETYPE
2字节
CFI≠1
CFI=1
E-RIF
控制
2字节
LTH<=2
LTH>2
E-RIF
描述符
2字节
(LTH-2)/2次
IPX标题
校验和
=FFFF
2
字节
分组
长度
2
字节
TC
1
字节
PT
1
字节
目标网络
4
字节
目标节点
6
字节
目标
插口
2
字节
源网络
4
字节
源节点
6
字节
源插口
2
字节

图3L

以太网MAC标题
DA
SA
8100
6字节
6字节
2字节
VLAN分组
TCI
2字节
LEN/
ETYPE
2字节
CFI=1
CFI=0
E-RTP
控制
2字节
LTH>2
LTH<=2
E-RTP
端述符
2字节
(LTH-2)个字
Etype=
8137
IP标题
校验和
2字节
分组长度
2字节
TC
1字节
PT
1字节
目标网络
4字节
目标节点
6字节
目标套接字
2字节
源网络
4字节
源节点
6字节
源套接字
2字节

图3K

802.3 MAC标题
DA
SA
长度
6字节
6字节
2字节
LLC LPDU
DSAP=
AA
SSAP=
AA
控制=
03
1字节
1字节
1字节
out
000000
3字节
Etype=
8137
2字节
IPX标题
校验
和
分组
长度
TC
PT
目标网络
目标节点
目标
套接字
源网络
源节点
源套
接字
2
字节
2
字节
1
字节
1
字节
4
字节
6
字节
2
字节
4
字节
6
字节
2
字节

图3J

802.3MAC标题
DA
SA
长度
6字节
6字节
2字节
LLC LPDU
DSAP=E0
SSAP=E0
控制=03
1字节
1字节
1字节
IPX标题
校验和
分组长度
TC
PT
目标网络
目标节点
目标套接字
源网络
源节点
源套接字
2字节
2字节
1字节
1字节
4字节
6字节
2字节
4字节
6字节
2字节

图3I

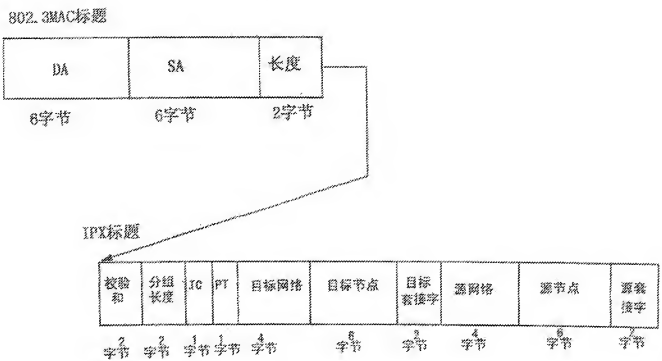

802.3MAC标题
DA
SA
长度
6字节
6字节
2字节
IPX标题
校验和
分组长度
TC
PT
目标网络
目标节点
目标套接字
源网络
源节点
源套接字
2字节
2字节
1字节
1字节
4字节
6字节
2字节
4字节
6字节
2字节

图3H

以太网MAC标题
DA
SA
8137
6字节
6节字
2字节
IPX标题
校验和
分组长度
TC
PT
目标网络
目标节点
目标套接字
源网络
源节点
源套接字
2字节
2字节
1字节
1字节
4字节
6字节
2字节
4字节
6字节
2字节

图3E

图3F

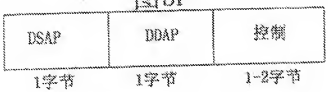


图3G

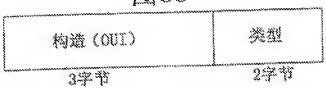


图3A

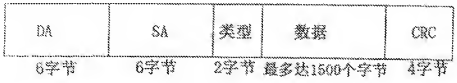


图3B

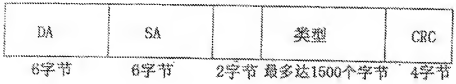


图3C

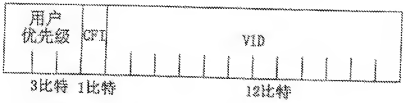


图3D

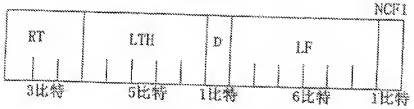


图2

EPC的框图

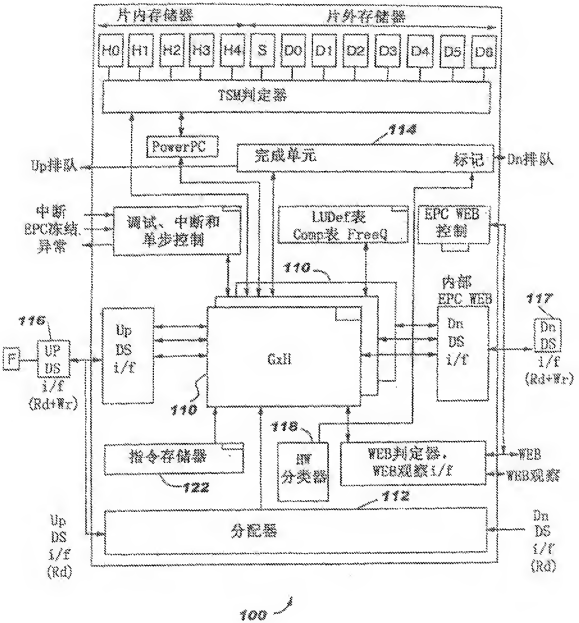
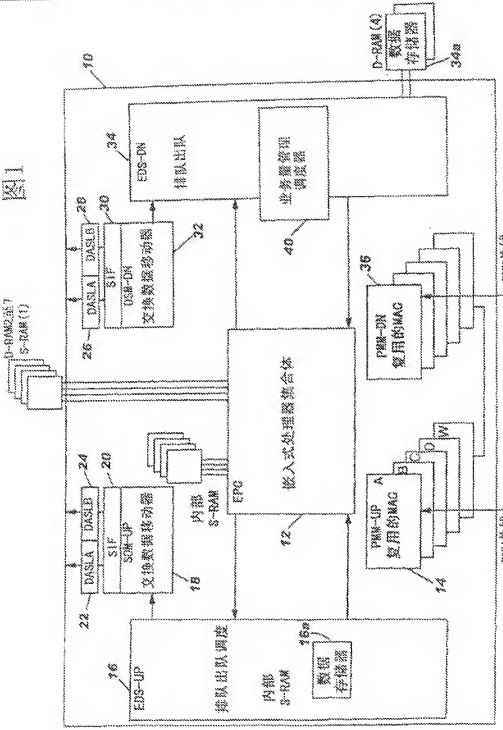

片内存储器
片外存储器
H0
H1
H2
H3
H4
S
D0
D1
D2
D3
D4
D5
D6
TSM判定器
PowerPC
114
完成单元
标记
Up排队
Dn排队
中断
EPC冻结
异常
调试、中断和
单步控制
LUDef表
Comp表 FreeQ
EPC WEB
控制
110
内部
EPC WEB
117
116
F
UP
DS
i/f
(Rd+Wr)
Up
DS
i/f
GxH
Dn
DS
i/f
Dn
DS
i/f
(Rd+Wr)
110
118
指令存储器
122
HW
分类器
WEB判定器,
WEB观察i/f
WEB
WEB观察
112
Up
DS
i/f
(Rd)
分配器
Dn
DS
i/f
(Rd)
100

图1


10
D-RAM (4)
数据存储器
34a
34
EDS-DN
排队出队
业务管理组调度器
40
D-RAM或者S-RAM (1)
26
DASLA
DASLB
SIF
DSM-DN
交换数据移动器
32
28
30
36
PMM-DN
复用的MAC
DMI总线
38
ENET PHY-ATM FRAMER
嵌入式处理器集合体
EPC
12
内部S-RAM
18
24
20
22
SIF
SDM-UP
交换数据移动器
PMM-UP
复用的MAC
14
A
B
C
W
排队出队调度
内部S-RAM
16a
数据存储器
16
EDS-UP

为，在正常运行中（没有刷新选择时），只有作为消息流的头的帧才会被完成单元考虑发送到循环装置。然而，由于各数据流都有各自的头，因此，一个数据流可能被阻塞，而其他数据可以继续进行处理并在没有中断或障碍以及没有干扰的情况下将完成的信息单元发送到循环装置。这对单个数据流被停止（例如处理器出问题或无法处理单个数据流的一个单元）而其他数据流不应被停止的情况尤其有用。否则，在该单个数据流阻塞被排除之前整个处理都将会停止。

当然，对熟悉相关技术的人员而言，纵观本优选实施方式的以上描述，并参照附图，还将看到本发明的许多修改。例如，实现分类器的硬件的实际类型可以有多种设计选择，并且一些所描述的具体选择取决于消息内容和消息的压缩方法以及需进行的处理。在不违背本发明的思想的前提下，还可以对系统的实现方式以及系统所能处理的消息结构作出许多修改。所存储的标记可以按其他方法而不根据消息的内容来产生，或者，可以只是由分配器所标识的数据流的连续编号。在不违背本发明的思想的前提下，还可以很好地采用本系统的许多其他修改方式和改进方式，并且，无需相应地使用其他相关的特性就能得到本发明的某些优点。因此，本实施方式的以上描述应当认为只不过是本发明的原理的说明，而并不局限于此。

数据流相关的其他标记存储器。

应当记住，当存在与单个处理器相关的两个标记存储器时，标记存储器之一表示存储在缓冲器（有时称为就绪 FCB 页）中的已完成或处理后的信息单元，该信息单元准备从处理集合体传送到适当的循环单元以用于传输，根据情况可以是上循环或是下循环，向上传送是指传送到接口设备，而向下传送是指传送回到数据传输网络。在这种情况下，处理器 0、处理器 4 和处理器 7 包括了在这两个相关的标记存储器中的数据。与各就绪 FCB 页相关的还有一个 UP 字段（表示这是向上页还是向下页），以及一个关于（在向下页情况下）帧是发向目标端口还是发向普通端口的指示符，这样可以判断是将向下页发送到向下目标端口循环装置还是发送到普通端口循环装置。如果处理器 0 的较早接收到的帧是标记存储器 M02，并且这既是数据流的头又是上循环装置为处理器集合体及其缓冲器输出的下一传输所选择的向上帧，那么，从就绪 FCB 页中取出 FCB 页及相关数据字段，从而将信息送到上循环装置。然后，切换第一标记指示符，以表示处理器 1 的其他标记存储器 M01 目前是第一个标记存储器，并且将标记存储器 M02 的有效字段 V 置为 0，表示这一标记存储器不再是激活的或有效的而相关的 FCB 页的有效字段 VF 被复位到 0。

本发明可以支持新数据流，而不必扰乱现有的数据流并且无需事先知道新数据流。代表新数据流（例如来自处理单元之一的关于其状态的消息）的分组只是与其标识数据一起被存储，而不必参考另一个数据流。如果它没有标识符，将不会与任一设置了 “无标记”字段的现有数据流（随时可以传输的消息）的标识符相匹配。

本发明还考虑到了刷新命令，以便将通过使系统接收完成的帧被接收时的次序处理完成的帧以按相应的次序处理数据流的操作覆盖，从而可以忽略标记字段的束缚（下一指针和在访问转发帧的循环装置之前给定帧是头的要求）。这可以通过将 “无标记字段”强加给 FCB 页来实现。

单个数据流将一直保持阻塞，直到处理完该数据流的头，这是因

两个标记存储器。这些处理器中的每一个处理器还与一个输出缓冲器（有时称为就绪 FCB 页）相关，用于处理后、正等待传输到三个所示的循环装置的帧。与每个标记存储器对相关的是第一标记存储器，以指示首先接收到哪个标记存储器（并且，当这两个标记存储器都有效时，第一标记代表处在就绪 FCB 页缓冲器的标记，而第二或后来接收到的标记代表当前在相应处理器中被处理的标记）。本图中示出了 5 个单独的数据流，尽管在任何给定时刻通过的数据流的个数取决于系统（尤其是其容量和网络业务量）并且随时间而变化。本例中，10 个处理器被标记为处理器 0-9，而标记存储器被标记为处理器 0 所用的存储器 M01 和 M02 至处理器 9 所用的标记存储器 M91 和 M92，但是处理器的个数是一个设计选择并可以变化（如果需要的话）。标识符为 A 的第一数据流在标记存储器 M01 中起动，如标记存储器 M01（以及处理器 0 的相应参考）表示一连串数据流的头的指示（H=1）所示。标记存储器 M01 的 “下一个”字段 N 指向标记存储器 M21，以表示处理器 2 正在处理与该数据流相关的下一信息单元。标记存储器 M21 的 “下一个”字段指向标记存储器 M52，它表示处理器 5 具有该数据流的下一部分。标记存储器 M52 的设置尾字段表示这是在 N 个处理器中当前正被处理的这一数据流的最后部分。本例中的数据流次序用从标记存储器 M01 指向标记存储器 M21 的箭头 A1 和从标记存储器 M21 指向标记存储器 M52 的箭头 A2 来表示，以说明数据流的单元之间的逻辑连接（这些箭头逻辑上表示 “下一个”字段中的指针，而物理上在实际实现中并不存在）。类似地，从标记存储器 M02 到标记存储器 M11 的数据流通过箭头 A3 来表示同一数据流的次序（尽管是一个与结合标记存储器 M01、M21 和 M52 所述不同的数据流）。第三数据流用与标记存储器 M31 和 M42 有关的箭头 A4 来表示，而第四数据流用标记存储器 M71 与 M72 之间的箭头 A5 来表示。最终，第五数据流在没有箭头的标记存储器 M41 中表示，这是因为，它是目前只包括单一标记存储器的数据流。这一标记存储器 M41 既是该数据流的头又是尾，且没有 “下一个”字段，这是因为，并不存在与这一

尾（直到接收到同一数据流的下一帧，此时将尾字段 T）。然后，在块 610 中，将这一标记与处理器中任一处理器正处理的当前标记进行比较（当然，其有效字段 V 必须是 1，表示这是个有效帧）。这一比较的结果是：当前标记等于进行中的一个标记，在这种情况下控制进至块 670，或者它与当前进行中的任一标记都不符，在这种情况下，控制进至块 630。如果有一个与当前标记之一相符，那么，该帧是现有数据流的一部分，因此，在块 670 中，将该数据流的前一终点的尾字段 T 复位（因此 T=0），并使该标记字段的 “下一个”字段指针指向当前帧的位置。然后，在块 680 中，头字段 H 被置为 0，表示当前帧不是数据流的头。如果当前帧的标记不等于当前所存储的任一标记，那么当前帧是一个新数据流，并且该当前帧是该新数据流的起点，于是在块 630 中将头字段 H 置为 1 以表示这种状态。在块 630 或块 680 中进行了设置适当标志（尤其是头字段 H 的标志）的处理后，将完成与现有数据流链接和设置字段或标志的处理。

在图 11 中，示出了用于处理或传送处理器输出的帧的过程。首先，在块 710 中，翻转第一字段指示符，使得，指针指向其他存储器。作为处理器的第一或下一字段。然后，在块 720 中，有效字段 V 被复位为 0，表示该数据不再有效（帧已被分配出去，并且该数据不表示当前被处理的帧）。块 725 测试尾字段 T 是否被置位（T=1），以表示这是特定数据流的尾帧。如果是，那么控制进至块 740，表示过程被完成。如果不是，那么，在块 730 中，找到连续的下一帧（通过 “下一个”字段中的指针），并使该帧的头比特或标志 H 被置位，以表示它是处理器中当前数据流的第一帧。然后，离开块 730，如块 740 所示，完成标志的设置。

图 12 举例示出了本发明的系统，说明了如上所述的具有图 10 中的逻辑的完成单元是如何容纳多个数据流的。N 个处理器以及分配器 112 和完成单元工作了一段时间，因此，图 12 表示完成单元的一部分（尤其是标记存储器）中所存储的数据的瞬间状态。如该图中所示，标记排队与众多的标记存储器连接，N 个处理器中的每个处理器都有

作或者处理后，正等待从处理集合体中输出的帧。在N个处理器的处理集合体中正被处理的每个数据流在这N个处理器中的某处有一个头或起点（即该数据流的最先被接收到的帧），并且该起点用其相关的标记字段单元的头字段H中的1标识为“头”。同样，这些处理器中的每个数据流当前在这N个处理器中还有一个尾帧，并且该尾帧因尾字段T中的1而被标识为尾。

有效字段V指示处理器是含有实际数据（因为它可能来自处理过程）（在有效字段中用1表示），还是不含实际数据（在有效字段V中用0表示）。当处理刚开始时，系统中没有实际或有效数据，因此，有效字段V被置为0，作为系统初始化的一部分。然后，随着数据从给定处理器的就绪FCB页510中读出，便将与该处理器的FCB页相应的有效字段V置为0，这表示该处理器不再有与该标记相应的有效信息（因为，FCB页中的信息已被传送到循环装置；尽管，由于该处理器本身可能正在处理一个不同的帧，该处理器可能在与该处理器相关的别的标记字段中还有有效信息）。“下一个”字段N表示与同一数据流中的下一帧相关的标记字段——与N个处理器相关的2N个标记字段中的另一个标记字段。标记排队480接收来自用于各输入信息单元或帧的分配器的关于给定帧已被分配的消息，以及该数据流的标识符和该帧所分配给的处理器的标识符。

图10示出了图4的标记排队480的流程。当一个帧从分配器112被分配给n个处理器之一时，在块600中，输入信息单元或帧的标记通过线路482发送到标记排队480，而n个处理器中的正在处理该帧的那个处理器的标识通过线路484发送。标记排队550的第一处理是在块602中判断对于第一标记字段指向的某个存储器而言有效字段V是否为1。如果有效字段V为1，那么所指向的存储器已被占用，于是应将数据存储到块606所指定的其他存储器中，否则，应当在块604中使用所指向的存储器。接着，在块650中，将该合适的存储器的有效字段V置为1，以表示在该存储器中存储了有效数据，然后，在块640中，使当前存储位置的尾指示符T置位以表示这是当前数据流的

图 8 示出了在处理输入帧和使用所述数据管理技术时所用的完成单元 114 的详细结构。本实施方式中所示的完成单元 114 与用于分配处理单元的输出（如处理后的信息单元）的多个循环装置（图 4 中未示出）进行通信。多个循环装置包括：一个上循环装置 450；和两个下循环装置，即一个标记为 460 的用于目标端口（少数常用的专门寻址的端口）的循环装置和一个标记为 470 的用于普通分配（发给除专门寻址的目标端口之外的其他端口的处理信息）的循环装置。

逻辑 "与"门 452、462、472 分别为循环装置 450、460、470 提供了选通。对于将帧提供给上循环装置 450 的 "与"门 452，输入是：作为 UP 帧的帧（来自与就绪 FCB 页 510 相关的块 UP），作为有效帧的帧（指示符 VF，表明是有效帧，传输就绪），在相关的帧标记字段中有效的标记字段（M01 至 M92），和与数据流的头（即最早的）帧相关的标记。

当帧被分配给给定的处理器时，分配器 112 将两条信息提供给标记排队 480——线路 482 上的帧的标记和线路 484 上的帧所分配给的处理器的身份。帧的标记识别该帧所属的数据流。在本优选实施方式中，这基于 MAC 加上源地址减去目标地址，其目标是为各个数据流提供唯一的标识符，这样，来自同一数据流的帧将具有相同的标记，而来自不同数据流的帧其标记或标识符将不同。

图 9 示出了用于存储与 N 个处理器中的每个处理器相关的信息的标记字段单元 500 的格式。N 个处理器中的每个处理器都有两个这样与其相关的标记字段，一个用于正被处理的帧，一个用于已被处理并正等待从处理集合体中输出的帧。处理后的准备要传送的帧保存在有时也称为就绪 FCB 页的存储器 510 中，这些存储器之一针对 N 个处理器中的每一个而存在。

标记字段单元 500 包括标记 L、"头"字段 H、"有效"字段 V、"尾"字段 T 和 "下一个"字段 N。标记 L 可从消息内容中得到，它表示各个数据流的唯一标识符。头字段 H 将当前正被 N 个处理单元处理的数据流或一连串相关的帧的起点标识为正被处理的工

MAC-SA+DA，或将其他消息格式中的 LID 与 MID 字段进行逻辑异或。

如图 7 所示，可以为每帧建立三种表或队列的存储。首先，规定处理后的帧的队列 400 来保存完成的任务（一个输出，或从处理给定帧的处理器处接收到的处理后的帧），因此对于每个处理器要求缓冲器或存储器空间可用于至少一个完成的帧（如帧 0 至帧 N），其中标记为 NPU-0 至 NPU-N 的处理器与各自的帧相联。当分配器 112 将一个帧发送给一个处理单元时，它将该帧的标识符发送给与各网络处理单元 NPU-0 至 NPU-N 相应的包括存储单元 0 至 n 的第二存储器或队列 410。当具有标识符或标记 m 的帧被发送给 NPU-0 时，那么，与 NPU-0 的标记相应的存储器 0 接收到所标识的 m 后进行存储，这表示 NPU-0 正在处理标识符为 m 的输入信息单元。可以理解，具有相同标识符 m 的后面的帧将属于相同数据流，而标识符或标记不同的帧将表示不同的数据流。因此，如果标记为 0 的输入信息单元被接收并分配给 NPU-1，那么将 0 记录在与 NPU-1 相应的存储器 1 中。于是，如果以后来自相同数据流（标记同样为 0）的第二输入信息单元被分配器 112 接收到并分配给处理器 NPU-N，那么，存储器 N 也存储标记 0，表示该信息单元被分配给了处理器 N。

第三存储器 420 包括存储当前正被 n 个处理单元处理的标记中的每一个标记。对于这些标记中的每一个标记，将存储所分配的处理器的标识符，并且，由于列表是依次的，因此，分配给特定消息流的第一处理器首先出现在存储器中。在这种情况下，对于标记 m，存储器 422 中的条目 0 表示 NPU-0 正在处理来自该流的输入信息单元，而对于标记 0，第一单元正被用存储器 424 表示的处理器 NPU-N 所处理而第二单元正被用存储器 426 表示的处理器 NPU-1 所处理。对于给定的流，要保持输入信息单元到达分配器时的次序，这样，同一数据流的后续传输可以与它被接收到时的次序相同，因此，将可以看到，标记存储器 424、426 使得这些 NPU 即处理单元按输入帧从网络处被接收到并被分配给 N 个处理器的次序列出。

Final CD 15802-3"，1998 年 2 月 20 日发布的 "IEEE Draft Standard 802.1Q/D9"； RFC 1700－1994 年 10 月 J.Reynolds 和 J.Postel 的分配号（该文献还可以从 http://www/isi.edu/rfc-editor/rfc.html 得到）；"IBM Token Ring Network Architecture Reference"；和 "IBM LAN Bridge and Switch Summary"，出版号为 SG24-5000-00 的版本 1.3，1996 年 1 月，具体参见第 1.1.1 节.

硬件分类器的设计可以有各种各样的方法，包括：使用多种通用软件工具之一来设计和制造硬件（或硅衬底的实际实现）结构中的逻辑电路设计方案，以及由逻辑电路设计者亲手通过传统设计方法来设计。本例中，所需的测试利用称为 VLSI 硬件定义语言（简称为 VHDL）的软件语言被编程，从而做成一个已知的软件（比如 IBM 销售的软件或 Synopsis 销售的软件）以实现具有所必需的门电路和逻辑电路的设计方案，从而完成硬件方式中的所需测试。另外，还有一些其他类似的设计系统可以很好地使用，因此，逻辑的设计者不必知道门电路的结构或其位置，只须知道其所需输入和测试以及输出的逻辑功能。

如上所述，在某些系统中，可能要求在本发明的处理系统中包括这样的性能，以便按数据流的帧被接收时的次序来转发处理后的数据流的帧，而与哪些处理器被指定用来处理每个帧无关。在这种系统中，分配器 112 在识别了可用的处理单元并将所接收的帧分配给这一用于处理的处理单元后，将针对该帧和被分配了该帧的处理单元产生并存储标识信息。

帧到达时一般具有标识信息，比如帧的消息编号（有时称为 MAC）以及源地址（有时称为 SA）和目标地址（有时称为 DA）。这种信息的位置和内容可能随消息的格式及其压缩技术而变化，但这一信息使得帧可以很好地通过该系统和交换机及路由器被发送到目标，并按合适的次序被组合成一个完整的消息，即使整个消息的长度大于单个帧。通常，一个消息的组成部分被称为数据流，而数据流的每一部分将包括相同的标识信息（比如 MAC、SA 和 DA）。分配器 112 单元分配给输入帧的实际标记（或标识信息）可以以多种方式来形成，比如

型的 LLC 即逻辑链路控制字段（例如图 3K 中所用的 AAAA03）。如果这一字段被认为是 SAP 字段之一，那么，设置这一 SAP 字段并在块 323 中保存协议信息，然后在块 325 中认为分类结束。如果这一字段是 SNAP 字段，那么，控制进至块 324，在此，得到 FISH3，并针对所识别的 ETYPE 分析 FISH3 的第 2-6 个字节。如果识别了 ETYPE，那么在块 323 中保存协议信息，然后在块 325 中退出。

如果在块 320 中判定第 13-14 个字节等于 8100，表示这是 IEEE 标准 802.1q 中所规定的虚拟局域网（VLAN），那么，在块 330 中保存 VLAN 的存在，然后，在块 340 中检查是否存在 CFI 字段。如果有，那么分类结束并且控制进至块 325。如果没有，那么，在块 350 中测试 FISH3 的第 1-2 个字节以判断它们是提供了已知的 ETYPE（如同块 320 中的测试）还是提供了长度（小于 0600H）。如果它们提供了 ETYPE，那么在块 323 中保存协议信息，然后控制进至块 325，在此认为分类结束。如果在块 350 中认为该字段不是 ETYPE，那么在块 325 中认为分类过程结束。如果块 350 中的测试提供了长度（小于 0600H），那么，在块 360 中针对已知的 SAP 测试第 3-5 个字节。如果它是 AAAA03，那么控制进至块 370，以针对已知的 ETYPE 确定第 6-10 个字节。

图 6 示出了一种改进型的硬件分类器，该改进特别针对图 4 中所示的单元。在图 6 中，硬件分类器包括了图 4 中的单元。但对指令控制逻辑电路 110c 有所改进，它不是包括单一的起点地址，而是包括了存储在指令堆栈 110d 中的一系列地址。这一指令堆栈包括了初始指令地址，随后还包括了当处理器到达分叉或分支时所需的其他地址，从而还避免了在后来的分支处进行测试或条件陈述。然后，这些起点地址按次序存储在堆栈中，并当需要分支指令时从堆栈中取出。

要获得关于各种协议或压缩技术的以太网消息的定义内容的进一步信息，读者可以查阅以太网帧结构的有关标准或参考指南。在了解以太网协议和压缩技术以及相应标准和选项时，可用的一些通用文献有：1997 年 11 月 24 日的 IEEE p802.1D/D15 的附录 C，“ISO/IEC

果硬件分类在线路 295 上被使能并且块 280 中没有确定不同的控制入口点，那么使用缺省入点；否则，使用来自表 280 的控制入口点。

来自硬件分类器 118 的线路 270、272（分别具有由硬件分类器辅助 118 所确定的分类标志和 L3 基地址）被输入到所分配的用于处理该帧的单个处理器 110，并被存储到与一个正在处理存储在数据存储器 110b 中的帧的处理单元相关的通用寄存器 110a 中。来自器件 295 的输出线 276 为这一特定类型的帧提供了指令存储器 122 的起始地址，即存储在指令控制逻辑电路 110c 中的数据。ALU（运算/逻辑单元）是处理单元 110 的一个部件。处理器 110 利用指令控制逻辑电路 110c 中的指令计数器从指令存储器 122 中取出指令。这样，根据硬件分类器辅助 118 所确定的协议和压缩方法，处理单元 110 用适合于正在处理的帧的指令集的起始地址进行了预处理，并且，设置合适的指示帧类型的标志，从而使处理器 110 可以利用正确的指令开始对帧进行处理。

图 5 示出了用于确定消息格式的分类的逻辑。这一逻辑从选定了 FISH2 的块 310 开始，然后，在块 320 中，测试该帧的第 13-14 个字节（即包括了帧中的类型信息的两个字节，该帧在类型前面包括了 6 个字节的目标地址 DA 和 6 个字节的源地址 SA）。如果这些字节符合 ETYPE0 或 ETYPE1 的内容，那么，该过程将在块 323 中通过设置一个适当的标志来标识协议信息，并在块 325 中结束过程。否则，如果类型块小于 0600H（十六进制），那么，该帧是以太网 802.3 的帧格式而不是以太网 V2.0DIX 的格式，并且该字段是长度字段而不是类型字段，于是，其处理在图 5 的左侧图中进行。如果这一类型块为 8100，那么，该帧是利用了 802.1q VLAN 支持的帧（参见例如图 3L、3M、3N、3O、3S 和 3T），于是，其处理在图 5 的右侧图中进行。如果类型字段是其他，那么，控制进至块 325，在此，认为分类结束，而无需记录任何协议信息，这是因为，该帧显然是未知协议。

如果在块 320 中判定第 13-14 个字节小于 0600H，那么，在块 322 中对第 15-17 个字节进行分析，以判断它们被认为是 SAP 字段还是类

助 118 对与输入信息单元（或帧）相关的 128 比特片段进行操作，这 128 比特片段有时称作 “FISH”，它可被分类器硬件辅助 118（也被各处理单元 110 之一）从分配器 112 所接收到。这一分类功能对一直到前 3 个 FISH（即与帧相关的前 384 比特，有时称为 FISH1、FISH2 和 FISH3 以便于相互区分这些 FISH）进行操作。第一个 FISH（FISH1）实际上不是所接收到的帧，而是一组与该帧有关的信息，比如，帧从哪个端口进来，缺省代码入口点 291，和是否利用本发明的硬件分类器进行帧分类的指示符 292（“是”或 “否”）。

在块 210 中，在帧中的不同位置来比较以太网的类型，以判断字段是否符合目前配置的协议，比如，第一以太网版本（如 IPx）或第二以太网版本（如 IPv4）。在块 220 中，判断 SAP（业务接入点）字段是否符合目前配置的协议，这仍是判断其是否如寄存器中所规定的值（例如，具体的存储值，指示协议类型）。该系统还在块 240 判断是否存在表示不同压缩类型的 SNAP 字段（诸如 “AAAA03”的特定字段），并在块 250 中检测消息中是否存在虚拟局域网（VLAN）使用。块 260 是分类控制，在被使能分类 292 使能时，该分类控制可以负责存储与帧相关的参数，并在线路 270、272、274 上提供一个指示协议类型、第 3 层指针和分类标志的输出。

各消息的控制入口点（处理的起点，指令存储器 122 的第一指令的地址）针对各种规定的格式可以事先被确定并存储在表 280 中。也就是说，对于 ETYPE=0 而无 VLAN，则控制入口点（起点地址）是指令存储器中的地址 122a，而对于 ETYPE=1 而无 VLAN，则控制入口点是地址 122b。类似地，对于 ETYPE=0 而有 VLAN 和 ETYPE=1 而有 VLAN，则各自的控制入口点（实际消息的处理的开始位置）分别是指令 122c 和 122d。对于具有 ERIF 字段的帧，处理将从指令 122f 开始，而对于未找到协议或压缩方法的缺省程序，处理将从指令 122f 开始。

无论在哪种情况，缺省控制入口点都包含在消息的 FISH1 中并在块 290 中被读取。然后，块 295 判断是否使用缺省控制入口点——如

VLAN 分组（与图 3M 中一样）；VLAN 分组（也具有与图 3M 一样的格式）；LLC LPDU（类似于参照图 3J 所示和所述的情况）；和 IPX 标题（如图 3H 中所示）。

图 3O 示出了基于具有 SNAP 和利用 802.1q 的 VLAN 支持的以太网 802.3 的 IPX 中的消息的结构或格式。它与图 3N 的格式类似，其中在 LLC LPDU 字段与 IPX 标题之间增加了 SNAP 字段。

图 3P 示出了基于以太网的 IPv4 的格式，其中，该消息包括了以太网 MAC 标题与 IPv4 标题。各字段的长度如图中所示。

图 3Q 示出了基于具有 802.2 的以太网 802.3 的 IPv4 的消息格式，图中示出了 MAC 标题，随后是 LLC LPDU，然后是 IPv4 标题。

图 3R 示出了基于具有 SNAP 的以太网 802.3 的 IPv4 帧的消息格式，其中，802.3 MAC 标题后面是 LLC LPDU，然后是 IPv4 标题（并带有可选的用于 UDP 或 TCP 尾部，如果适用的话）。

图 3S 示出了基于具有 802.1q VLAN 支持的以太网的 IPv4 的消息格式。这一格式具有 IPv4 和如 802.1q VLAN 支持的其他例子中所示的 VLAN 分组的特性。

图 3T 示出了基于具有 802.1q VLAN 支持的以太网 802.3（具有 802.2）的 IPv4 的消息格式，它将具有 802.2 的 802.3 上的 IPv4 的特性与 VLAN 分组的消息特性相结合。

在图 3H 至 3T 的每一图中，底部那行代表帧或消息的第 3 层（即 L3）部分，并且，由于在消息的 L3 部分前面的内容的大小的变化，因此，根据消息的类型（协议和压缩方法），消息的 L3 部分将在不同位置开始。尽管要求对 L3 消息进行处理（忽视压缩），然而，在多协议和多压缩系统中，可能难以找到 L3 消息的起点。此外，由于多个处理器 110 之一针对帧所执行的指令取决于帧协议和压缩方法的类型，因此，希望有某种东西（在这种情况下是硬件分类器辅助 118）来为处理器进入指令存储器 122 提供一个到达正确的起始指令的指针。

图 4 示出了图 2 中用单元 118 表示的分类器硬件辅助连同指令存储器 122 的选定部分和多个处理单元 110 之一的框图。分类器硬件辅

字节的源地址 SA 和目标地址 DA，随后是 2 个字节的类型 8137（表示该帧具有 IPX 格式）。然后，IPX 标题包括所示的这样一些组成部分，即：2 个字节的校验和，2 个字节的分组长度，1 个字节的 TC，1 个字节的 PT，4 个字节的目标网络，6 个字节的目标节点，2 个字节的目标套接字，4 个字节的源节点，6 个字节的源节点，和 2 个字节的源套接字。

图 3I 示出了基于以太网 802.3 的专用版本的 IPX 的消息格式（有时称为 Novell 格式），它包括以太网 802.3 MAC 标题，其中，在第三字段中规定了该消息的长度（而不是图 3H 中所示的以太网上的 IPX 中的类型）。这一格式中的校验和根据其协议被设为 “FFFF”。

图 3J 示出了基于具有 802.2 的以太网 802.3 的 IPX，其中，该消息包括了被 802.2 的 LLC LPDU 字段所分开的 MAC 标题和 IPX 标题（与图 3H 中所示一样）。

图 3K 示出了基于具有 SNAP 的 802.3 的 IPX 帧的格式，其中，就象参照图 3J 所示的格式那样，该消息包括了 802.3 MAC 标题，随后是 LLC LPDU 字段，最终是 IPX 标题。安排在 LLC LPDU 部分与 IPX 标题之间的是 SNAP 字段，用于指示 OUI 和 ETYPE8137。

图 3L 示出了基于具有 802.1q VLAN 支持的以太网的 IPX 的格式，其中，类型字段表示为 8100，而 VLAN 分组安排在以太网 MAC 标题与 IPX 标题之间（IPX 标题具有与以上参照图 3H、3J 和 3K 所述相同的格式）。VLAN 分组包括 2 个字节的 TCI 字段和 2 个字节的长度 LEN 或 e-类型字段，然后是 e-rif 控制字段和不定个数的 e-rif 描述符字段，其个数由公式(LEN-2)/2 来指定。

图 3M 示出了基于利用 802.1q VLAN 支持的以太网 802.3（专用）的 IPX 的格式。类型字段为 8100，而 VLAN 分组与前面图 3L 中的 VLAN 例子中的情况类似。IPX 标题与前面图 3I 中的 802.3 专用帧中所示的情况类似，其中，校验和字段设为等于 “FFFF”。

图 3N 示出了基于采用具有 VLAN 支持的以太网 802.3 的 IPX 的帧的帧结构。它包括：802.3 MAC 标题，其中类型为 8100，表示有

图 3D 示出了一种嵌入式 RIF（即 E-RIF）格式，它用于某些以太网协议消息格式，而且遵循 IEEE 标准 802.1q。在这种格式中，路由类型 RT 用前 3 比特来表示，长度 LTH 用随后的 5 比特来表示（指示整个 E-RIF 部分的字节的长度，包括 E-RIF 路由控制和 E-RIF 路由描述符），而路由描述符方向 D 用 1 比特来表示，（通常，“0”表示按前向次序遍历路由描述符，而在某些专门发送的帧中，用“1”表示路由描述符处于反向次序）。E-RIF 格式包括 6 比特最大帧指示符和 1 比特非规范格式指示符（NCFI）。路由类型 RT 可以是 00X、01X、10X 或 11X，以指示该帧分别是专门发送的帧、透明帧、全程探测帧或跨越树探测帧。根据以太网的 IEEE 802.3 标准，最大帧 LF 字段等于或小于 1470 个字节。NCFI 指示所指定的 MAC 地址是非规范形式的（如果为 0）还是规范形式的（如果为 1）。

图 3E 示出了 E-RIF 路由描述符格式，它包括 12 比特局域网标识 LAN ID 和 4 比特桥接号（“桥接号”）。E-RIF 路由描述符格式字段在工业上是众所周知的，此用法遵循这种字段的标准。

图 3F 和图 3G 示出了以太网消息中所用的 LLC 格式的组成部分，包括图 3F 中的 802.2 LPDU 格式和图 3G 中的一般 SNAP 格式。图 3F 的 LPDU 格式包括 1 个字节（8 比特）目标业务接入点 DSAP、1 个字节源业务接入点 SSAP 和 1 至 2 个字节控制字段“控制”（包括命令、响应、序号和轮询/最终比特）。这种情况下，业务接入点为 6 比特加上 U 比特和最终比特（单独的 1 比特用于目标业务接入点，而 C 比特用于源的命令/响应指示符）。图 3G 示出了 SNAP 格式，它包括用于指示构造（构造的唯一标识符，即 OUI）的 3 个字节和用于指示在因特网标准 0002 情况下分配给该格式的类型的 2 个字节。类型字段的例子有 IP 的 0800、IPX 的 8137、ARP 的 0806、RARP 的 8035、802.1q VLAN 的 8100、IPv6 的 86DD、Appletalk 的 80DB 和 Appletalk AARP 的 80F3。

图 3H 示出了基于以太网格式的 IPX 中的消息的格式，它包括以太网 MAC 标题和 IPX 标题，其中，以太网 MAC 标题具有各为 6 个

字节，而 CRC 尾部被规定为 4 个字节。通常，消息的剩余部分（即 “数据”）可具有最多达 1500 个字节的任意长度，不过，从后面可以看到，某些类型的以太网为了得到其他好处对这一灵活性进行限制。源地址 SA 可以指示该消息是一个预定给网络中的某个节点的单一网络地址的单个消息，或者它是一个多点传播或一个广播消息。多点传播消息是发给网络中的一组节点的，而广播则是发给所有站的。指示 “类型”的块是 16 比特，它用于识别所采用的较高层协议。每一所寄存的以太网协议都给定一个唯一的类型码，即一个总是大于以太网 802.3 长度字段的长度字段中的最大值的值，以使该字段可以共存。数据字段其长度通常为 46-1500 个字节，假定，在将数据传送给 MAC 层之前，较高的层可以保证满足 46 个字节的最小字段。长度大于所允许的帧长度的消息必须分割成多个长度小于数据字段的最大允许长度的消息。

图 3B 示出了一种一般以太网格式的变形，它称为 IEEE 802.3 以太网格式。该格式除了类型字段用长度字段 LEN 取代之外，其他与图 1 中的一般以太网消息格式的格式类似，该长度字段共 16 比特，用于指示随后的数据字段（不包括任何填充）的长度。该标准强行规定分组的最小长度为 64 个字节，因此，数据字段 “数据”必须至少为 46 个字节。如果数据字段 “数据”的实际数据少于 46 个字节，那么，MAC 层必须将一些空间补足块（place saver）（填充字符）添加到 LLC 数据字段中，以便在通过网络发送分组之前达到最小容量。然而，长度字段是指不含填充字符的长度，这使得接收系统可以识别并不考虑所添加的填充字符。

图 3C 示出了一种用于以太网消息的标记控制信息格式，尤其遵循 IEEE 标准 802.1q。它包括 3 比特用户优先级、1 比特规范格式指示符即 CFI 和 12 比特 VID 即虚拟 LAN（即 VLAN）标识符。虚拟 LAN 或局域网是一组节点的标识（这些节点已通过将地址定义为包括 VLAN 而被识别为虚拟局域网），从而使得这些物理上不相关的节点可以在逻辑上相关，并作为一个组来寻址，而不是单独来寻址。

图12
UP RR
450
逻辑电路
452
DN TPQ RR
460
逻辑电路
462
DN GEN. RR
470
逻辑电路
472
280
标记排队
处理器
582
标记
584
分配器
112
510
逻辑FOB页0
逻辑FOB页N
处理器0
处理器N
M01
M11
M21
M31
M41
M51
M61
M71
M81
M91
M02
M12
M22
M32
M42
M52
M62
M72
M82
M92
标记A
标记B
标记C
标记D
标记E
HVTTF
A1
A2
A3
A4
A5
A6
F0

图11

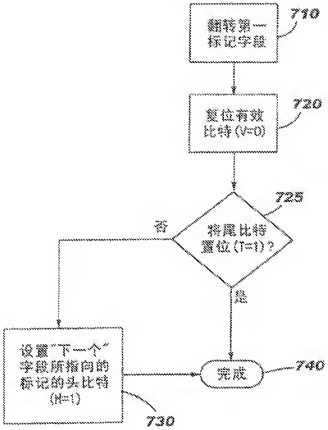
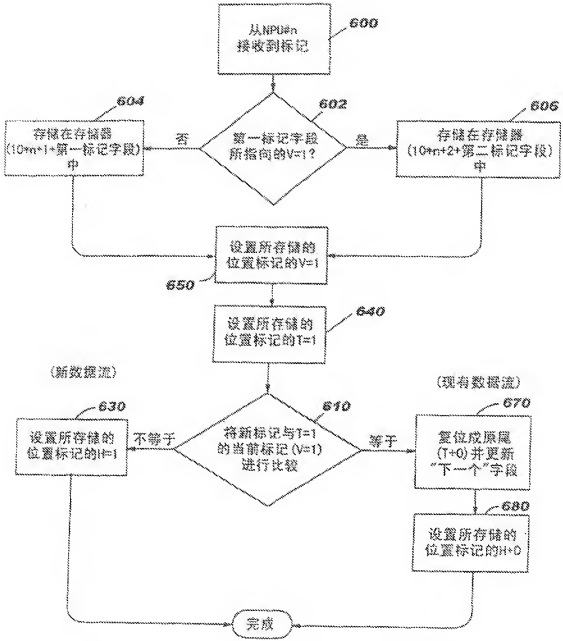

翻转第一标记字段
710
复位有效比特(V=0)
720
将尾比特置位(T=1)?
725
否
是
设置"下一个"字段所指向的标记的头比特(H=1)
730
完成
740

图10


从NPUn
接收到标记
600
第一标记字段
所指向的V=1?
602
否
是
存储在存储器
(10*n+1+第一标记字段)
中
604
存储在存储器
(10*n+2+第二标记字段)
中
606
设置所存储的
位置标记的V=1
650
设置所存储的
位置标记的T=1
640
(新数据流)
(现有数据流)
将新标记与T=1
的当前标记 (V=1)
进行比较
610
不等于
等于
设置所存储的
位置标记的H=1
630
复位成原隘
(T+0)并更新
"下一个"字段
670
设置所存储的
位置标记的H+0
680
完成

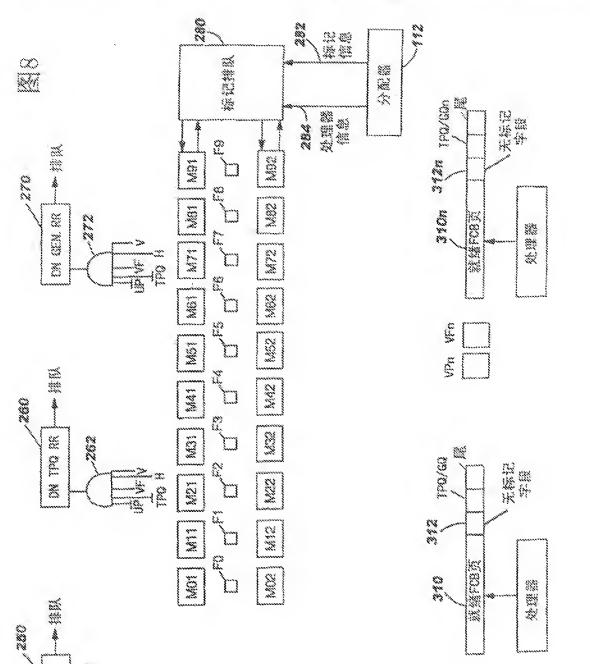


图8
UP RR
排队
DN TPG RR
排队
DN GEN RR
排队
标记排队
分配器
处理器
跳线FCB页
处理器
跳线FCB页
处理器
TPG/GQ
尾
无标记字段
TPG/GQn
尾
无标记字段
处理器信息
标记信息
UP
VF
V
H
TPG H
VFn
VPn
M01
M11
M21
M31
M41
M51
M61
M71
M81
M91
M02
M12
M22
M32
M42
M52
M62
M72
M82
M92
F1
F2
F3
F4
F5
F6
F7
F8
F9
260
252
262
270
272
280
282
264
112
310
312
310n
312n

图7

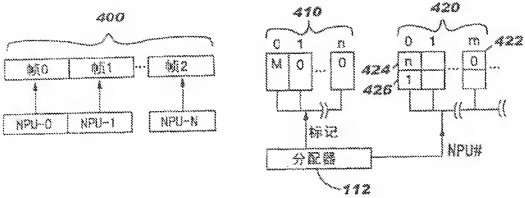

400
帧0
帧1
帧2
NPU-0
NPU-1
NPU-N
410
0 1 n
M 0 0
420
0 1 m
422
424
426
n
1
标记
分配器
NPU#
112

图9

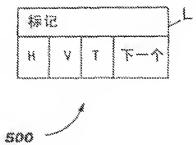

标记
L
H
V
T
下一个
500

图6

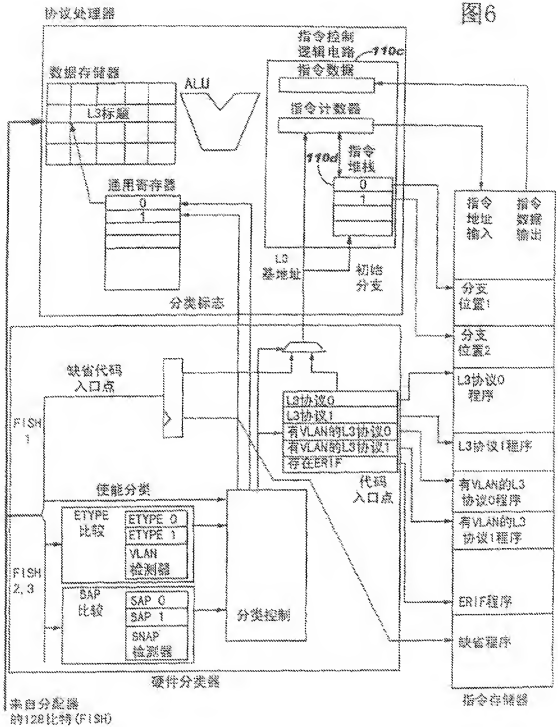

协议处理器
指令控制
逻辑电路
110c
数据存储器
ALU
指令数据
L3标题
指令计数器
110d
指令
堆栈
0
1
通用寄存器
0
1
指令
地址
输入
指令
数据
输出
L3
基地址
初始
分支
分类标志
分支
位置1
分支
位置2
缺省代码
入口点
L3协议0
L3协议1
有VLAN的L3协议0
有VLAN的L3协议1
存在ERIF
L3协议0
程序
FISH
1
L3协议1程序
代码
入口点
有VLAN的L3
协议0程序
使能分类
有VLAN的L3
协议1程序
ETYPE
比较
ETYPE 0
ETYPE 1
VLAN
检测器
FISH
2, 3
SAP
比较
SAP 0
SAP 1
SNAP
检测器
分类控制
ERIF程序
缺省程序
硬件分类器
指令存储器
来自分配器
的128比特(FISH)

图5

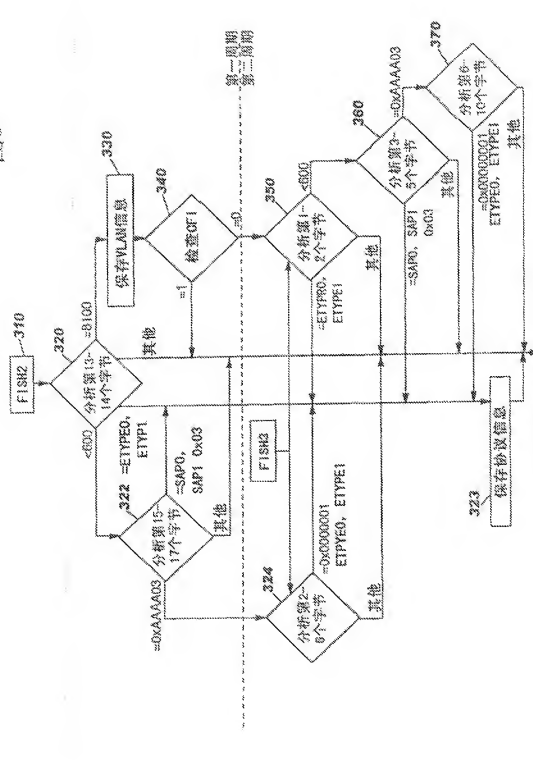

FISH2
310
320
分析第13-14个字节
<600
=8100
其他
保存VLAN信息
330
检查CFI
340
=1
=0
第一周期
第二周期
分析第15-17个字节
322
=0xAAAAAA03
=ETYPE0, ETYP1
=SAP0, SAP1 0x03
其他
FISH2
分析第2-6个字节
324
=0x0000001 ETYPE0, ETYPE1
其他
分析第1-2个字节
350
<600
=ETYPE0, ETYPE1
其他
分析第3-5个字节
360
=0xAAAAAA03
=SAP0, SAP1 0x03
其他
分析第6-10个字节
370
=0x0000001 ETYPE0, ETYPE1
其他
保存协议信息
323
分类结束
325

图 4

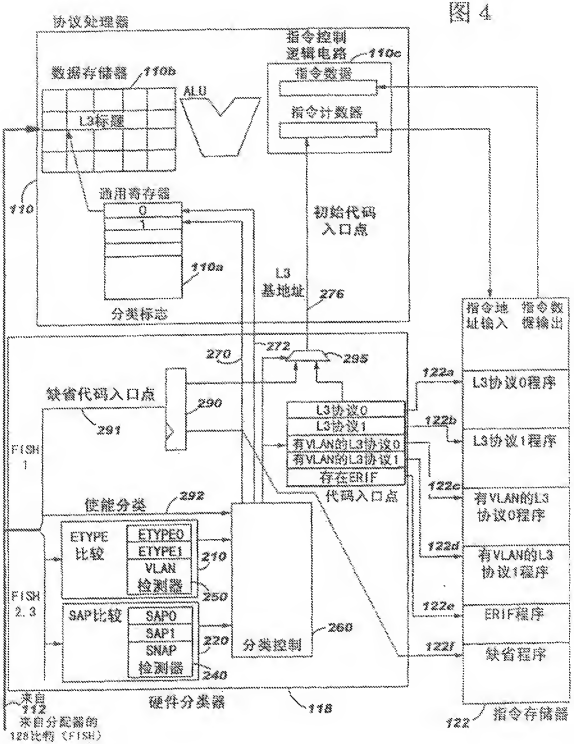

协议处理器
指令控制
逻辑电路 110c
指令数据
指令计数器
数据存储器 110b
ALU
L3标题
通用寄存器
0
110
110a
初始代码
入口点
L3
基地址 276
分类标志
指令地址输入
指令数据输出
270
272
295
缺省代码入口点
290
291
L3协议0
L3协议1
有VLAN的L3协议0
有VLAN的L3协议1
存在ERIF
代码入口点
122a
122b
122c
122d
122e
122f
L3协议0程序
L3协议1程序
有VLAN的L3协议0程序
有VLAN的L3协议1程序
ERIF程序
缺省程序
FISH 1
使能分类 292
ETYPE比较
ETYPE0
ETYPE1
VLAN检测器
210
250
FISH 2.3
SAP比较
SAP0
SAP1
SNAP检测器
220
240
分类控制
260
来自 112
来自分配器的128比特（FISH）
硬件分类器 118
122 指令存储器

图31

802.3 MAC帧头
DA
SA
8100
6字节
6字节
2
字节
VLAN分组
TCI
2字节
LEN/
ETYPE
2字节
CFI=1
E-RIF
控制
2字节
CFI≠1
LTH<=2
LTH>2
E-RIF
描述符
2字节
(LTH-2)/2次
LLC LPDU
DSAP=
06
SSAP=
06
控制=
03
1字节
1字节
1字节
IPv4标题
如果是UDP
或TCP
源端口
目标
端口
2
字节
2
字节

图3S

以太网MAC帧头
DA
SA
8100
6字节
6字节
2
字节
VLAN分组
TCI
LEN/
ETYPE
OFI≠1
OFI=1
E-RIF
控制
LTH<=2
LTH>2
E-RIF
指示符
2字节
(LTH-2)/2次
Etype=
800
IPv4标题
如果是UDP
或TCP
ver
hlen
服务
类型
总
长度
标识
标志
偏移
TTL
协议
校验和
源IP地址
目标IP
地址
可选项/
填充
源端口
目标
端口
比特
字节
1
字节
2
字节
3
比特
字节
13
比特
字节
1
字节
1
字节
2
字节
4
字节
4
字节
0-n
字节
2
字节
2
字节

Method and system for frame and protocol classification

Publication number: CN1433543

Publication date: 2003-07-30

Inventor: BASS BRIAN MITCHELL (US); CALVIGNAC JEAN LOUIS (US); AL GORDON TAYLOR DAVIS ET (US)

Applicant: IBM (US)

Classification:

- International: ***H04L12/56; H04L12/28; H04L12/46; H04L29/06; H04L12/56; H04L12/28; H04L12/46; H04L29/06;*** (IPC1-7): G06F9/46; H04L29/06

- European: H04L29/06

Application number: CN20008018837 20001221

Priority number(s): US20000479027 20000107; US20000479028 20000107

Also published as:

WO0150259 (A1)
MXPA02005419 (A)
EP1244964 (A0)
CA2385339 (A1)
EP1244964 (B1)

more >>

Report a data error here

Abstract not available for CN1433543
Abstract of corresponding document: **WO0150259**

A system and method of frame protocol classification and processing in a system for data processing (e.g., switching or routing data packets or frames). The present invention includes analyzing a portion of the frame according to predetermined tests, then storing key characteristics of the packet for use in subsequent processing of the frame. The key characteristics for the frame (or input information unit) include the type of layer 3 protocol used in the frame, the layer 2 encapsulation technique, the starting instruction address, flags indicating whether the frame uses a virtual local area network, and the identity of the data flow to which the frame belongs. Much of the analysis is preferably done using hardware so that it can be completed quickly and in a uniform time period. The stored characteristics of the frame are then used by the network processing complex in its processing of the frame. The processor is preconditioned with a starting instruction address and the location of the beginning of the layer 3 header as well as flags for the type of frame. That is, the instruction address or code entry point is used by the processor to start processing for a frame at the right place, based on the type of frame. Additional instruction addresses can be stacked and used sequentially at branches to avoid additional tests and branching instructions. Additionally, frames comprising a data flow can be processed and forwarded in the same order in which they are received.